一、进程与线程的基础区别：

1、进程是资源分配的最小单位，线程是程序执行的最小单位（资源调度的最小单位）

2、进程有自己的独立地址空间，每启动一个进程，系统就会为它分配地址空间，建立数据表来维护代码段、堆栈段和数据段，这种操作非常昂贵。

而线程是共享进程中的数据的，使用相同的地址空间，因此CPU切换一个线程的花费远比进程要小很多，同时创建一个线程的开销也比进程要小很多。

3、线程之间的通信更方便，同一进程下的线程共享全局变量、静态变量等数据，而进程之间的通信需要以通信的方式（IPC)进行。不过如何处理好同步与互斥是编写多线程程序的难点。

4、但是多进程程序更健壮，多线程程序只要有一个线程死掉，整个进程也死掉了，而一个进程死掉并不会对另外一个进程造成影响，因为进程有自己独立的地址空间。

2.

进程——资源分配的最小单位，线程——程序执行的最小单位。

# 线程进程的区别体现在4个方面：（高级）

1、因为进程拥有独立的堆栈空间和数据段，所以每当启动一个新的进程必须分配给它独立的地址空间，建立众多的数据表来维护它的代码段、堆栈段和数据段，这对于多进程来说十分“奢侈”，系统开销比较大，而线程不一样，线程拥有独立的堆栈空间，但是共享数据段，它们彼此之间使用相同的地址空间，共享大部分数据，比进程更节俭，开销比较小，切换速度也比进程快，效率高，但是正由于进程之间独立的特点，使得进程安全性比较高，也因为进程有独立的地址空间，一个进程崩溃后，在保护模式下不会对其它进程产生影响，而线程只是一个进程中的不同执行路径。一个线程死掉就等于整个进程死掉。

2、体现在通信机制上面，正因为进程之间互不干扰，相互独立，进程的通信机制\*\*\*\*相对很复杂，譬如管道，信号，消息队列，共享内存，套接字等通信机制，而线程由于共享数据段所以通信机制很方便。

3、体现在CPU系统上面，线程使得CPU系统更加有效，因为操作系统会保证当线程数不大于CPU数目时，不同的线程运行于不同的CPU上。

4、体现在程序结构上，举一个简明易懂的列子：当我们使用进程的时候，我们不自主的使用if else嵌套来判断pid，使得程序结构繁琐，但是当我们使用线程的时候，基本上可以甩掉它，当然程序内部执行功能单元需要使用的时候还是要使用，所以线程对程序结构的改善有很大帮助。

# 什么情况下使用进程个线程：

1、需要频繁创建销毁的优先使用线程；因为对进程来说创建和销毁一个进程代价是很大的

2、线程的切换速度快，所以在需要大量计算，切换频繁时用线程，还有耗时的操作使用线程可提高应用程序的响应

3、因为对CPU系统的效率使用上线程更占优，所以可能要发展到多机分布的用进程，多核分布用线程

4、并行操作时使用线程，如C/S架构的服务器端并发线程响应用户的请求

5、需要更稳定安全时，适合选择进程；需要速度时，选择线程更好

        因为我的项目中需要对数据段的数据共享，可以被多个程序所修改，所以使用线程来完成此操作，无需加入复杂的通信机制，使用进程需要添加复杂的通信机制实现数据段的共享，增加了我的代码的繁琐，而且使用线程开销小，项目运行的速度快，效率高。

       如果只用进程的话，虽然安全性高，但是对代码的简洁性不好，程序结构繁琐，开销比较大，还需要加入复杂的通信机制，会使得我的项目代码量大大增加，切换速度会变的很慢，执行效率降低不少。。。

# 4..进程和线程的关系：

1、一个线程只能属于一个进程，而一个进程可以有多个线程，但至少有一个线程。线程是操作系统可识别的最小执行和调度单位。

2、资源分配给进程，同一进程的所有线程共享该进程的所有资源。 同一进程中的多个线程共享代码段(代码和常量)，数据段(全局变量和静态变量)，扩展段(堆存储)。但是每个线程拥有自己的栈段，栈段又叫运行时段，用来存放所有局部变量和临时变量。

3、处理机分给线程，即真正在处理机上运行的是线程。

4、线程在执行过程中，需要协作同步。不同进程的线程间要利用消息通信的办法实现同步。

# 5.简单的理解

做个简单的比喻：进程=火车，线程=车厢  
线程在进程下行进（单纯的车厢无法运行）  
一个进程可以包含多个线程（一辆火车可以有多个车厢）  
不同进程间数据很难共享（一辆火车上的乘客很难换到另外一辆火车，比如站点换乘）  
同一进程下不同线程间数据很易共享（A车厢换到B车厢很容易）  
进程要比线程消耗更多的计算机资源（采用多列火车相比多个车厢更耗资源）  
进程间不会相互影响，一个线程挂掉将导致整个进程挂掉（一列火车不会影响到另外一列火车，但是如果一列火车上中间的一节车厢着火了，将影响到所有车厢）  
进程可以拓展到多机，进程最多适合多核（不同火车可以开在多个轨道上，同一火车的车厢不能在行进的不同的轨道上）  
进程使用的内存地址可以上锁，即一个线程使用某些共享内存时，其他线程必须等它结束，才能使用这一块内存。（比如火车上的洗手间）－"互斥锁"  
进程使用的内存地址可以限定使用量（比如火车上的餐厅，最多只允许多少人进入，如果满了需要在门口等，等有人出来了才能进去）－“信号量”

6.